REMARKS

Applicants respectfully traverse and request reconsideration.

Claims 2-14 and 15-17 stand rejected under 35 U.S.C. §102(b) as being anticipated by Bonola. This is a new ground of rejection. The Bonola reference is directed to a different structure and operation from that of Applicants' claimed invention. For example, Bonola is directed to a distributed execution of mode mismatched commands in multiprocessor computer systems. As described, if a multimode processor encounters a command or instruction that it cannot execute without shifting operating modes, it will look for an alternate processor present in the computer system to instead handle the mode mismatch command. The originating processor suspends thread execution until the alternate processor completes execution. When complete, the alternate processor downloads the results to a common memory and signals its status to the originating processor. As such, the Bonola teaches halting execution of one of a number of other processors in a multiprocessor system until its execution is completed by another processor. As such, processing time can be wasted on a per-command basis when one processor does not understand a particular command without shifting a mode of operation.

In contrast, Applicants claim (see claim 15) a different structure wherein, for example, a homogenous set of processors are used to process differing functional programs so that different processors can carry out different tasks. In contrast, Bonola does not appear to teach a homogenous processing system since if it was homogenous, all of the processors could process the commands and there would not be a need for another processor to process a mismatched command. In any event, Applicants claim a different system since Applicants' system, among other things, has the differing multiprocessors perform different functional programs wherein the functional programs allow the processors to perform the tasks of at least graphics image

processing, video processing and audio processing and communications processing. The processing of the tasks using functional programs allows the differing types of tasks to be run, for example, in parallel without interrupting another processor unlike that taught in the Bonola reference. Also, the functional programs are carried out by the available processing resources as opposed to a single mismatch mode command as taught in Bonola. Accordingly, Applicants respectfully submit that the claims are in condition for allowance.

The dependent claims add additional novel and non-obvious subject matter.

As to claim 17, the office action cites to Bonola, column 9, lines 24-27 as allegedly teaching a plurality of input/output devices coupled to the plurality of processors that process program code configured to perform the plurality of tasks. However, the cited portion merely refers to a single slave processor that handles execution of a mode mismatch command encountered by a host processor. The one "task" that appears to be performed is a mode mismatch command execution by a single processor. Accordingly, this claim is also in condition for allowance.

In addition, the office action cites column 7, lines 22-25; 42-52 as allegedly teaching a plurality of input/output devices coupled to a plurality of processors to an input/output interface wherein a portion of the plurality of processors interact with a first and second input/output device. However, the cited portion describes again the control passing from a slave to a host CPU when a mode mismatch command is encountered. There is no discussion of interacting with first input and second input/output devices by the multiple processors nor is there a discussion of causing a second portion of the plurality of processors to emulate a specific microprocessor instruction set (office action cited to column 8, lines 11-18). For example, the cited portion does not refer to emulating an instruction set, but instead refers to a CPU loading

"parameters necessary to successfully carry out the mode mismatch command on the selective slave CPU". In the example given, a disk service routine is loaded into a selective slave CPU's general purpose register. There is no mention of microprocessor instruction set emulation in the cited portion and as such, this claim is also in condition for allowance.

The dependent claims add additional novel and non-obvious subject matter.

Accordingly, Applicants respectfully submit that the claims are in condition for allowance and that a timely Notice of Allowance be issued in this case. The Examiner is invited to contact the below-listed attorney if the Examiner believes that a telephone conference will advance the prosecution of this application.

Respectfully submitted,

Date: 8/3/05

By: C Reckamp
Christopher J. Reckamp
Reg. No. 34,414

VEDDER, PRICE, KAUFMAN &
KAMMHOLZ, P.C.
222 N. LaSalle Street
Chicago, IL 60601
(312) 609-7500
FAX: (312) 609-5005